

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-168744

(43) 公開日 平成7年(1995)7月4日

(51) Int.Cl. ⁴	識別記号	序内整理番号	F 1	技術表示箇所
G 0 6 F 12/00	5 0 1 B	8944-5B		
	5 9 1	9366-5B		
9/44	5 3 0 S	9193-5B		
		9194-5L	G 0 6 F 15/ 40	3 8 0 E
		9194-5L	15/ 401	3 4 0 Z

審査請求 未請求 請求項の数2 OL (全 13 頁) 最終頁に続く

(21) 出願番号 特願平5-316454

(22) 出願日 平成5年(1993)12月16日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 田中 哲郎

東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

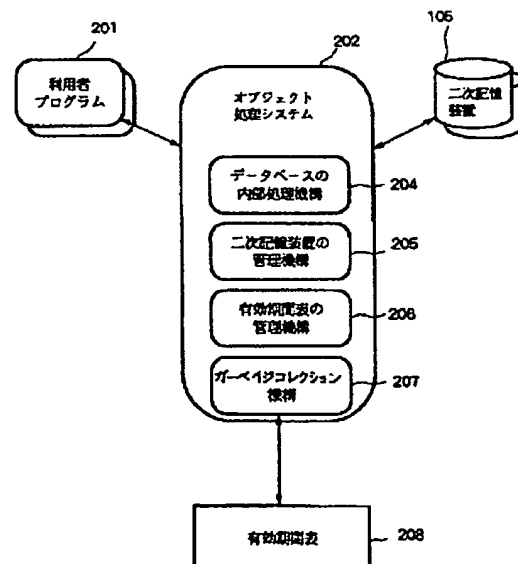
(74) 代理人 弁理士 大塚 康徳 (外1名)

(54) 【発明の名称】 情報管理装置及びその制御方法

(57) 【要約】

【目的】 装置或はシステムにかかる負担を軽減し、不要なオブジェクトの増加を抑制することを可能にする。

【構成】 オブジェクト(情報)を二次記憶装置105に登録する場合、そのオブジェクトの有効期間を設定する。そして、そのオブジェクトを特定するオブジェクト識別子OIDとその有効期間を管理する有効期間表208に追加記憶する。オブジェクトのガーベージ処理を行う場合には、その有効期間表208に記憶された内容に基づく、有効期間をすぎたオブジェクトについては、二次記憶装置105から削除する。



【特許請求の範囲】

【請求項 1】 情報を所定の記憶装置に複数個記憶し、管理する情報管理装置において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定手段と、登録対象の情報を特定する識別情報と当該登録対象の情報の有効期間を対応付けて記憶する記憶手段と、該記憶手段によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある登録情報の削除を行う削除手段とを備えることを特徴とする情報管理装置。

【請求項 2】 情報を所定の記憶装置に複数個記憶し、管理する情報管理装置の制御方法において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定工程と、登録対象の情報を特定する識別情報と当該登録対象の情報の有効期間を対応付けて記憶する記憶工程と、該記憶工程によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある登録情報の削除を行う削除工程とを備えることを特徴とする情報管理装置の制御方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は情報管理装置及びその制御方法、詳しくはデータベースにおけるオブジェクトのガーベージ処理を行う情報管理装置及びその制御方法に関するものである。

【0002】

【従来の技術】通常、オブジェクト指向データベースのガーベージコレクション機能は、データベースシステムに専用の処理機能を設けて、それをバックグラウンドまたはフォアグラウンドで処理を行ない、不要になった永続オブジェクト（ガーベージ）を発見し、その永続オブジェクト（例えば文書や画像等）の情報をハードディスクや光磁気ディスク等の大容量二次記憶装置から解放し、データベースの管理リストから削除するようにしている。

【0003】

【発明が解決しようとする課題】しかしながら上記従来例ではオブジェクト処理システムのガーベージコレクション機能がガーベージを発見し解放及び削除を行なうように作用するため次のような問題がある。

【0004】（1）ガーベージコレクションのための処理がシステムの負担になる。

【0005】（2）複合オブジェクトの要素になりうる永続オブジェクトが、ガーベージかどうかを判定するために、該複合オブジェクトの要素が他の複合オブジェクトの要素になっているかどうか検査する必要がある。

【0006】（3）永続オブジェクトから構成される複合オブジェクトがガーベージかどうかを判定する途中に、複合オブジェクトの要素及び構成が変更されないことを保証するために、トランザクション処理を行なう必

要がある。

【0007】（4）ガーベージコレクションがシステムで自動的に行なわれる場合、ガーベージが消去される時間を利用者が知ることができない。

【0008】更には、オブジェクト処理システムのオブジェクトに有効期間を指定できないため、次のような欠点があった。

【0009】（5）ある一定の期間にのみ有効な永続オブジェクトに対して有効期間を指定できない。

【0010】（6）ある一定の期間に不要な永続オブジェクトに対しては、該期間の前に明示的に削除する必要がある。

【0011】（7）永続オブジェクトを作成／消去するための二次記憶及びシステムの表への変更を行なう時刻を利用者が指定できない。

【0012】

【課題を解決するための手段】本発明はかかる問題点を鑑みなされたものであり、装置或はシステムにかかる負担を軽減し、不要なオブジェクトの増加を抑制することを可能にする情報管理装置及びその制御方法を提供しようとするものである。

【0013】この課題を解決するため本発明の情報管理装置は、以下に示す構成を備える。すなわち、情報を所定の記憶装置に複数個記憶し、管理する情報管理装置において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定手段と、登録対象の情報を特定する識別情報と当該登録対象の情報の有効期間を対応付けて記憶する記憶手段と、該記憶手段によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある登録情報の削除を行う削除手段とを備える。

【0014】また、本発明の情報管理装置の制御方法は以下に示す工程を備える。すなわち、情報を所定の記憶装置に複数個記憶し、管理する情報管理装置の制御方法において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定工程と、登録対象の情報を特定する識別情報と当該登録対象の情報の有効期間を対応付けて記憶する記憶工程と、該記憶工程によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある登録情報の削除を行う削除工程とを備える。

【0015】

【作用】かかる本発明の構成或は工程において、ある情報を登録しようとする場合には、その登録対象の情報の有効期間を設定する。そして、その登録対象の識別情報を設定された有効期間の対を記憶保持しておく。そして、この記憶された情報を元に、システムの実時間と比較し、無効期間になっている情報については削除する。

【0016】

【実施例】以下図面に従って本発明に係る実施例を詳細に説明する。

【0017】図 1 は、実施例の情報処理装置の構成を示

サブブロック図である。

【0018】同図において、101はキーボードやマウス等の入力装置、102は装置全体を制御する中央演算処理装置（以下、CPUという）、103はシステムで使用するプログラムやデータをロードする主記憶装置（以下、RAM）、104はCPU102の起動時に実行されるブートプログラム及びフォントデータ等を記憶している読み出し専用メモリ（ROM）である。105は、データベースのデータやデータベース処理プログラムが記憶されている二次記憶装置（ここでは、ハードディスク装置とする）であり、このプログラムはRAM103にロードされ、実行されることになる。106は、端末やプリンタ等の出力装置である。

【0019】図2は、上記構成におけるデータベースプログラムがRAM103にロードされている場合の構成概念図である。

【0020】同図において、201は利用者プログラム（データベースに対して処理要求を行う）、202はオブジェクト処理システム（データベースの内部処理機構204、二次記憶装置の管理機構205、有効期間表の管理機構206、ガーベジコレクション管理機構207から構成される）、208は2次記憶装置105内に記憶保持されている有効期間表（オブジェクトと有効期間の対応表）であり、詳細は後述する。

【0021】上記構成において、オブジェクト処理システム202及び利用者プログラム201を構成するのは、図1におけるCPU102、RAM103等である。

【0022】図3は、図2における有効期間表208の一例を示している。同図において、“オブジェクト識別子（OID）”はシステム内のオブジェクトを一意的に識別するための情報である。また、“有効期間”は対応するオブジェクトの有効期間を示す時刻情報であり、いつからいつまでの間で有効かを示す情報を保持している。つまり、この期間内であれば、システムはそのオブジェクトの存在を保証し、その期間外のオブジェクトは、ガーベジオブジェクトとみなし、2次記憶装置105から開放し、削除することを示している。

【0023】例えば、オブジェクト識別子“OID1”のオブジェクトは平成元年4月17日12時30分00秒から平成2年9月10日12時30分00秒までが有効期間であり、該有効期間において該オブジェクトの情報が二次記憶装置203中に保存されていることを示している。

【0024】また、オブジェクト識別子“OID4”のオブジェクトは有効期間がすべての期間で有効であり、明示的に消去しない限り二次記憶装置203中に記憶保持され、ガーベジコレクションの対象にはならないことを示している。

【0025】尚、OID2、OID3については、上記

説明から容易に推察されよう。

【0026】図2の有効期間表208の情報は、図2の二次記憶装置203に永続オブジェクトのデータとは別に保存しても良いし、表全体の情報をある一つの永続オブジェクトとして二次記憶装置203に保存してもよい。また、各永続オブジェクトの中の情報として各永続オブジェクト毎に二次記憶装置203に保存してもよい。

【0027】また、有効期間表208の情報は、最初の時刻を省略して、消去する時刻のみを指定するようにしても良い。この場合は、新たなオブジェクトの情報を二次記憶装置203中に書き込む（登録）時刻を指定することはできない。

【0028】さて、上記構成において、例えば、新たなデータ（オブジェクト）をシステムに登録する際には、オブジェクトに対する公知の付加情報（キーワードや作成者やその他の情報）を付けるが、この時点で、ユーザはそのオブジェクトに対する有効期間の入力（もし必要なら）を指示し、システムに登録依頼を指示する。

【0029】図4はかかる登録依頼を受けたシステムの処理手順を示している。

【0030】永続オブジェクトの作成要求が発生すると（ステップS401）、オブジェクトの有効期間を指定しているかどうかチェックする（判断ステップS402）。オブジェクトの有効期間が指定されていない場合（NOの場合）は、システムの規定値を利用する（ステップS403）。例えば、システムの規定値を作成時からある一定の期間を該オブジェクトの有効期間とすることもできる。また例えば、システムの規定値を永久に削除しないような有効期間（ $-\infty \sim +\infty$ ）とすることもできよう。これらの規定値は、システム的环境設定でユーザ（特に許されたユーザのみ）が自由に設定するようにしても良いし、ユーザが個々の規定値を設定するようにしても良い。

【0031】次に、過去に発生しているオブジェクト識別子OIDの番号に基づいて、新たなOID識別子を得る（ステップS404）。そして、そのオブジェクト識別子OIDと上記有効期間の組を有効期間表208に追加登録する（ステップS405）。

【0032】次に、永続オブジェクトのもつオブジェクト情報を二次記憶装置203中に書き込む（ステップS406）。

【0033】最後に、永続オブジェクトの作成要求元に返値を返す（ステップS407）。この返り値は、例えば2次記憶装置105にオブジェクトを記憶するだけの容量が確保されたかどうか等、処理が正常に進んだかどうかを示す情報である。

【0034】以上の結果、実施例のデータベースにおいては、原則として個々のオブジェクトにはOIDとその有効期間情報が作成されることになる。

【0035】次に、実施例におけるガーベジコレクション処理内容を図5のフローチャートに従って説明する。

【0036】まず、最初に、上記ガーベジコレクション機構207はガーベジになりうる永続オブジェクトを一つ選択する(ステップS501)。例えば、上記オブジェクト処理システム202が管理するオブジェクト識別子OIDから成る表(上記データベースの内部処理機構204が管理する表)を検索し、オブジェクト識別子OIDを順番に一つずつ選択してもよい(ケース1)、上記有効期間表208から直接オブジェクト識別子OIDを順番に一つずつ選択しても良い(ケース2)。

【0037】次に、当該永続オブジェクトのオブジェクト識別子OIDから上記有効期間表208を検索し、そのオブジェクトの有効期間を読み出す(ステップS502)。

【0038】そして、当該永続オブジェクトの有効期間と、システムが保持している現在の時刻(日付も当然含まれる)を比較し、そのオブジェクトが有効期間外であるか内であるかを判断する(ステップS503)。

【0039】有効期間外であると判断した場合(YESの場合)は、該当する永続オブジェクトを消去すると共に、有効期間表208の該当箇所も削除する。

【0040】有効期間を過ぎていない場合(NOの場合)はそのまま終了する。

【0041】以上の如く、本実施例によれば、個々のオブジェクトが自身の有効期間を示す情報を持ち、個々のオブジェクトを特定する情報と有効期間の対で構成される専用の情報(ファイル)を持つことで、システムに係る負担が小さくなり、不要なオブジェクトをすばやく、且つ、簡単に探し出すことが可能になる。しかも、不用意にガーベジオブジェクトが増加することもなくなる。

【0042】尚、上記例では、有効期間表208には、オブジェクトIDとその有効期間の対を記憶させたが、オブジェクト名とその存在位置を示す情報を含めるようにしても良い。この場合、有効期間表208の情報量は上記例と比較して多少多くなるが、オブジェクトを探し出す処理を高速にできる。

【0043】<他の実施例の説明>

【実施例2】オブジェクトの有効期間を図2の有効期間表208を利用するのではなく、各永続オブジェクトのデータ中に有効期間を指定するものである。オブジェクト処理システムのガーベジコレクション機構207が各永続オブジェクトから有効期間を読みだし、有効期限を過ぎたオブジェクトのみをガーベジコレクションの対象とするものである。ガーベジコレクション機構207は、最初に有効期間を調査して、有効期限内のオブジェクトはガーベジとは見なさず次のオブジェクトの

調査に移動するので、上記実施例と比較して処理速度は多少劣るものの、ガーベジコレクション処理を軽減することができる。

【0044】【実施例3】図4において、オブジェクトの有効期間を指定しない場合、システムの既定値を明示的な「有効期間なし」とするものである。

【0045】例えば、「有効期間なし」の場合、有効期間表208には何も書き込まないようにもできる。この場合、有効期間表208が大きくなるのを抑制することも可能になる。さらに、ガーベジコレクション機構207がガーベジの対象として最初に有効期間表208からオブジェクト識別子OIDを選択する場合(上記「ケース2」)、「有効期間なし」のオブジェクトは自動的にガーベジの対象から除外されるので、ガーベジの対象から除外したいオブジェクトは「有効性なし」として、作成することができる。

【0046】また例えば、「有効期間なし」の場合、上記有効期間表208には「有効期間なし」を示す情報のみを書き込む様にもできる。ガーベジコレクション機構207が「有効期間なし」の永続オブジェクトをガーベジの対象とするかどうかは、ガーベジコレクション処理の時点で決めることができるようになる。この場合は、明示的にオブジェクトの有効期間を設定しなくても、ガーベジコレクション処理の時点でガーベジかどうか決まることになる。

【0047】【実施例4】図4において、オブジェクトの有効期間を指定しない場合、各クラスの既定値を利用しても良い。例えば、有効期間を指定せずにオブジェクトを作成すると作成時からある一定の期間が該オブジェクトの有効期間となる。各クラス毎に既定値を変えることができるので細かい制御が可能になる。クラスの既定値をクラス変数として可変にできる。また、クラスオブジェクトを永続オブジェクトにして二次記憶装置203に保存し、クラスオブジェクト自体の有効期間が過ぎると、該クラスのインスタンスを全てガーベジ扱いにするようにもできる。また、クラスオブジェクトの有効期間を該クラスのインスタンス全ての有効期間を含む期間になるよう設定すると、該クラスのインスタンス全てがガーベジ扱いになると同時にクラスオブジェクトもガーベジ扱いになる。

【0048】【実施例5】クラスのバージョンを複数にしてスキーマを扱うデータベースにおいて、各クラスの差分情報及び最初のバージョンの情報のみを記憶している場合、あるバージョンのクラスオブジェクトの有効期間が過ぎた時点において該クラスオブジェクトを削除する際に、該クラスオブジェクトのバージョン以前の古いバージョンのクラスオブジェクトをマージし、スキーマの整合性を保つようにしても良い。

【0049】【実施例6】実施例5において、クラスオブジェクトの有効期間の代わりにクラスバージョン番号

を使用し、バージョンがある番号を越えた時点で該バージョン以前のクラスオブジェクトをマージし、不要なクラスオブジェクト及び不要なクラスのインスタンスを消去しても良い。最大のバージョン番号は、各スキーマ毎に指定できる。指定しない場合は、システムの既定値にもできるし、各スキーマ毎の既定値にもできる。

【0050】「実施例 7」実施例 5 においてあるクラスの有効期間が過ぎた時点 t において各クラスオブジェクトを削除する際に、該クラスオブジェクトのバージョンよりも古いバージョンのクラスオブジェクト全てについて有効期間を検査し全ての有効期間が t 以前の場合のみ、マージ及び削除を行なうものである。

【0051】「実施例 8」複合オブジェクトにおいて、複合オブジェクトの有効期間を指定する際に、該複合オブジェクトの要素であるオブジェクト全てについて有効期間を検査し、全ての要素の有効期間及び該複合オブジェクトの有効期間を含む有効期間 I を計算し、全ての要素の有効期間及び該複合オブジェクトの有効期間を該有効期間 I に設定しても良い。このようにして、複合オブジェクトの全ての要素の有効期間及び該複合オブジェクトの有効期間は等しくなるため、ガーベージ判定の際に、該複合オブジェクトの要素であるオブジェクト全てについて有効期間を検査する必要がなくなり、複合オブジェクトと全ての要素を同時に消去することができるようになる。

【0052】【実施例 9】実施例 8 において、複合オブジェクトの有効期間を指定していない場合に、該複合オブジェクトの要素であるオブジェクト全てについて有効期間を検査し、全ての有効期間を含む有効期間を該複合オブジェクトの有効期間としても良い。

【0053】【実施例 10】実施例 8 において、複合オブジェクトに要素オブジェクトを登録する際に、該要素オブジェクトの有効期間を検査し、全ての有効期間を含む有効期間を該複合オブジェクト及び該要素オブジェクトの有効期間としても良い。

【0054】【実施例 11】ある一定の期間毎にデータベースの情報を二次記憶装置 203 に書き込むシステムの場合、永続オブジェクトの有効期間を書き込み時点以前に指定して該オブジェクトを作成しようとしたとき、該永続オブジェクトを主記憶装置 103 にのみ存在する一時的オブジェクトとして作成し、二次記憶装置 203 および有効期間表 208 には書き込まないようにしても良い。このようにして無駄な書き込み処理を節約できるようになる。

【0055】【実施例 12】二次記憶装置 203 を複数個持つシステムの場合、オブジェクトの有効期間毎に異なる二次記憶装置 203 に記憶し、ある期間を過ぎた時点で二次記憶装置 203 全ての情報を消去するようにしても良い。例えば、物理的に二次記憶装置 203 を初期化することもできる。また、二次記憶装置 203 の情報

を消去せずに単に全て無意味な情報として上書き可能とすることもできる。この例では、システム全体で「二次記憶装置の識別子、有効期間」からなる T を用意し、二次記憶装置 203 毎に有効期間表 208 を用意し、オブジェクトを二次記憶装置 203 に保存する際に、該表 T から有効期間 $I1$ を読みだし、該オブジェクトの有効期間 $I2$ が該有効期間 $I1$ に含まれる場合は、該二次記憶装置に書き込み、含まれない場合は、該有効期間 $I1$ 及び $I2$ を含む有効期間 $I3$ が最小になるような $I1$ を計算し、対応する二次記憶装置にオブジェクトを書き込み、該表 T の $I1$ を $I3$ に変更する。このようにして、カーページ処理機構は該表 T から有効期間を読み出し、二次記憶装置単位で消去することができるようになる。この例では、二次記憶装置単位で消去するので、有効期間表 208 も同時に消去される。

【0056】【実施例 13】実施例 12 において、二次記憶装置 203 を複数個のパーティションに分割して使用するシステムの場合二次記憶装置単位の代わりにパーティション単位で、各オブジェクトの有効期間毎に異なるパーティションに記憶しある期間を過ぎた時点で、該パーティション全ての情報を消去するものである。また、パーティションの情報を消去せずに単に全て無意味な情報として上書き可能とすることもできる。

【0057】本発明は、オブジェクト指向データベースについて述べたが、他のデータベースについても考えられる。

【0058】尚、本発明は、複数の機器から構成されるシステムに適用しても 1 つの機器から成る装置に適用しても良い。また、本発明は、システム或は装置にプログラムを供給することによって達成される場合にも適用できることはいうまでもない。

【0059】

【発明の効果】以上説明したように本発明によれば、システムにかかる負担を軽減し、不要なオブジェクト（情報）の増加を抑制することが可能になる。

【0060】

【図面の簡単な説明】

【図 1】実施例の情報処理装置の構成を示すブロック構成図である。

【図 2】実施例の情報処理装置の構成概念図である。

【図 3】実施例における有効期間表の内容を示す図である。

【図 4】実施例における永続オブジェクトの作成及び登録の手続きの処理手順の一部を示すフローチャートである。

【図 5】ガーベージコレクション機構が永続オブジェクトの中からガーベージを選択し消去する処理の内容を示すフローチャートである。

【符号の説明】

1 使用者プログラム（データベースに対して処理要求

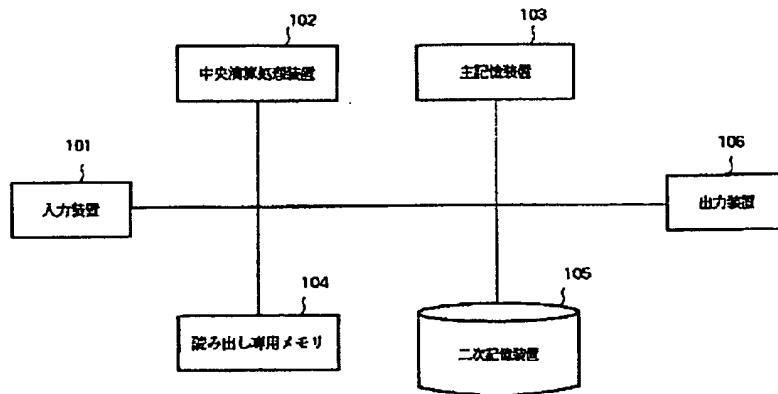
をおこなう)

* ページの消去を行なう)

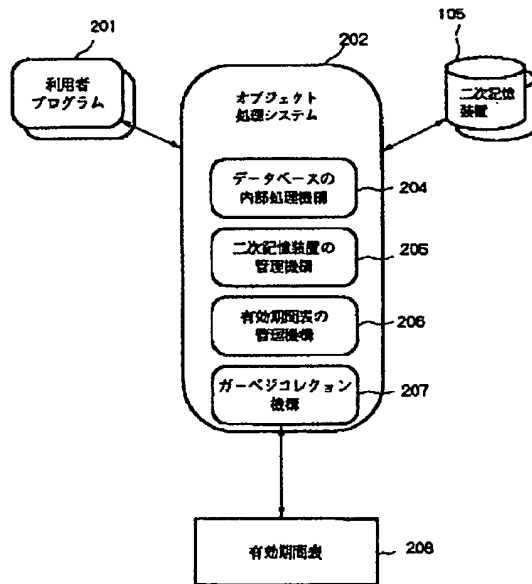
2 オブジェクト処理システム (データベースの内部処理を行なう。二次記憶装置及び有効期間を管理し、ガー*

3 二次記憶装置 (データベースのデータを保存する)
4 有効期間表 (オブジェクトと有効期間の対応表)

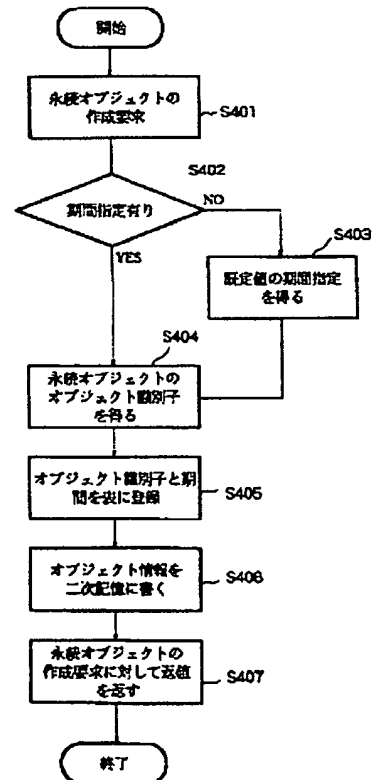
【図 1】



【図 2】



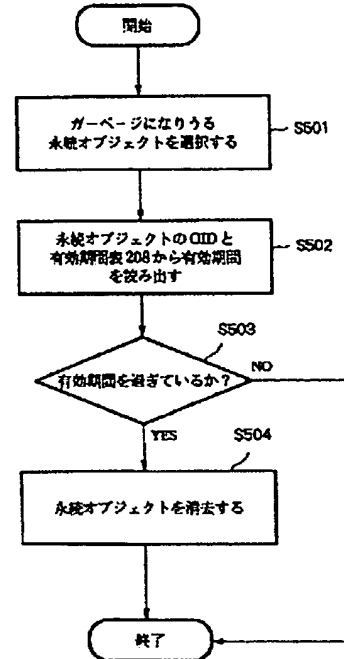
【図 4】



【図3】

オブジェクト識別子	有効期間
OID1	平1年5月1日12時30分00秒～平3年7月1日12時30分00秒
OID2	-∞～平2年6月30日12時30分00秒
OID3	平9年5月1日22時30分00秒～+∞
OID4	-∞～+∞

【図5】



【手続補正書】

【提出日】平成6年5月9日

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】全文

【補正方法】変更

【補正内容】

【書類名】明細書

【発明の名称】情報管理装置及びその制御方法

【特許請求の範囲】

【請求項1】 情報を所定の記憶装置に複数個記憶し、管理する情報管理装置において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定手段と、登録対象の情報を特定する識別情報と当該登録対象の情報の有効期間を対応付けて記憶する記憶手段と、該記憶手段によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある登録情報の削除を行う削除手段とを備えることを特徴とする情報管理装置。

【請求項2】 情報を所定の記憶装置に複数個記憶し、管理する情報管理装置の制御方法において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定工程と、登録対象の情報を特定する識別情報と当該登録対象の情

報の有効期間を対応付けて記憶する記憶工程と、該記憶工程によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある登録情報の削除を行う削除工程とを備えることを特徴とする情報管理装置の制御方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は情報管理装置及びその制御方法、詳しくはデータベースにおけるオブジェクトのガーベージ処理を行う情報管理装置及びその制御方法に関するものである。

【0002】

【従来の技術】通常、オブジェクト指向データベースのガーベージコレクション機能は、データベースシステムに専用の処理機能を設けて、それをバックグラウンドまたはフォアグラウンドで処理を行ない、不要になった永続オブジェクト（ガーベージ）を発見し、その永続オブジェクト（例えば文書や画像等）の情報をハードディスクや光磁気ディスク等の大容量二次記憶装置から解放し、データベースの管理リストから削除するようにしている。

【0003】

【発明が解決しようとする課題】しかしながら上記従来例ではオブジェクト処理システムのガーベージコレクシ

ョン機能がガーベージを発見し解放及び削除を行なうように作用するため次のような問題がある。

【0004】(1) ガーベージコレクションのための処理がシステムの負担になる。

【0005】(2) 複合オブジェクトの要素になりうる永続オブジェクトが、ガーベージかどうかを判定するために、該複合オブジェクトの要素が他の複合オブジェクトの要素になっているかどうか検査する必要がある。

【0006】(3) 永続オブジェクトから構成される複合オブジェクトがガーベージかどうかを判定する途中に、複合オブジェクトの要素及び構成が変更されないことを保証するために、トランザクション処理を行なう必要がある。

【0007】(4) ガーベージコレクションがシステムで自動的に行なわれる場合、ガーベージが消去される時間を利用者が知ることができない。

【0008】更には、オブジェクト処理システムのオブジェクトに有効期間を指定できないため、次のような欠点があった。

【0009】(5) ある一定の期間にのみ有効な永続オブジェクトに対して有効期間を指定できない。

【0010】(6) ある一定の期間に不要な永続オブジェクトに対しては、該期間の前に明示的に削除する必要がある。

【0011】(7) 永続オブジェクトを作成/消去するための二次記憶及びシステムの表への変更を行なう時刻を利用者が指定できない。

【0012】

【課題を解決するための手段】本発明はかかる問題点に鑑みなされたものであり、装置或はシステムにかかる負担を軽減し、不要なオブジェクトの増加を抑制することを可能にする情報管理装置及びその制御方法を提供しようとするものである。

【0013】この課題を解決するため本発明の情報管理装置は、以下に示す構成を備える。すなわち、情報を所定の記憶装置に複数個記憶し、管理する情報管理装置において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定手段と、登録対象の情報を特定する識別情報と当該登録対象の情報の有効期間を対応付けて記憶する記憶手段と、該記憶手段によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある登録情報の削除を行う削除手段とを備える。

【0014】また、本発明の情報管理装置の制御方法は以下に示す工程を備える。すなわち、情報を所定の記憶装置に複数個記憶し、管理する情報管理装置の制御方法において、情報を記憶登録する際、当該情報の有効期間を設定する期間設定工程と、登録対象の情報を特定する識別情報と当該登録対象の情報の有効期間を対応付けて記憶する記憶工程と、該記憶工程によって記憶されている情報とシステムの実時刻とを比較し、無効期間にある

登録情報の削除を行う削除工程とを備える。

【0015】

【作用】かかる本発明の構成或は工程において、ある情報を登録しようとする場合には、その登録対象の情報の有効期間を設定する。そして、その登録対象の識別情報を設定された有効期間の対を記憶保持しておく。そして、この記憶された情報を元に、システムの実時刻と比較し、無効期間になっている情報については削除する。

【0016】

【実施例】以下図面に従って本発明に係る実施例を詳細に説明する。

【0017】図1は、実施例の情報処理装置の構成を示すブロック図である。

【0018】同図において、101はキーボードやマウス等の入力装置、102は装置全体を制御する中央演算処理装置（以下、CPUという）、103はシステムで使用するプログラムやデータをロードする主記憶装置（以下、RAM）、104はCPU102の起動時に実行されるブートプログラム及びフォントデータ等を記憶している読み出し専用メモリ（ROM）である。105は、データベースのデータやデータベース処理プログラムが記憶されている二次記憶装置（例えば、ハードディスク装置や光磁気ディスク装置等）であり、このプログラムはRAM103にロードされ、実行されることになる。106は、端末やプリンタ等の出力装置である。

【0019】図2は、上記構成におけるデータベースプログラムがRAM103にロードされている場合の構成概念図である。

【0020】同図において、201は利用者プログラム（データベースに対して処理要求を行う）、202はオブジェクト処理システム（データベースの内部処理機構204、二次記憶装置の管理機構205、有効期間表の管理機構206、ガーベージコレクション機構207から構成される）、208は二次記憶装置105内に記憶保持されている有効期間表（オブジェクトと有効期間の対応表）であり、詳細は後述する。

【0021】上記構成において、オブジェクト処理システム202及び利用者プログラム201を構成するのは、図1におけるCPU102、RAM103等である。

【0022】図3は、図2における有効期間表208の一例を示している。同図において、“オブジェクト識別子（OID）”はシステム内のオブジェクトを一意的に識別するための情報である。また、“有効期間”は対応するオブジェクトの有効期間を示す時刻情報であり、いつからいつまでの間で有効を示す情報を保持している。つまり、この期間内であれば、システムはそのオブジェクトの存在を保証し、その期間外のオブジェクトは、ガーベージオブジェクトとみなし、二次記憶装置105から開放し、削除することを示している。

【0023】例えば、オブジェクト識別子“OID1”のオブジェクトは平成元年4月17日12時30分00秒から平成2年9月10日12時30分00秒までが有効期間であり、該有効期間において該オブジェクトの情報が二次記憶装置105中に保存されていることを示している。

【0024】また、オブジェクト識別子“OID4”のオブジェクトは有効期間がすべての期間で有効であり、明示的に消去しない限り二次記憶装置105中に記憶保持され、ガーベジコレクションの対象にはならないことを示している。

【0025】尚、OID2、OID3については、上記説明から容易に推察されよう。

【0026】図2の有効期間表208の情報は、図2の二次記憶装置105に永続オブジェクトのデータとは別に保存しても良いし、表全体の情報をある一つの永続オブジェクトとして二次記憶装置105に保存してもよい。また、各永続オブジェクトの中の情報として各永続オブジェクト毎に二次記憶装置105に保存してもよい。

【0027】また、有効期間表208の情報は、最初の時刻を省略して、消去する時刻のみを指定するようにしても良い。この場合は、新たなオブジェクトの情報を二次記憶装置105中に書き込む（登録）時刻を指定することはできない。

【0028】さて、上記構成において、例えば、新たなデータ（オブジェクト）をシステムに登録する際には、オブジェクトに対する公知の付加情報（キーワードや作成者やその他の情報）を付けるが、この時点で、ユーザはそのオブジェクトに対する有効期間の入力（もし必要なら）を指示し、システムに登録依頼を指示する。

【0029】図4はかかる登録依頼を受けたシステムの処理手順を示している。

【0030】永続オブジェクトの作成要求が発生すると（ステップS401）、オブジェクトの有効期間を指定しているかどうかチェックする（判断ステップS402）。オブジェクトの有効期間が指定されていない場合（NOの場合）は、システムの規定値を利用する（ステップS403）。例えば、システムの規定値を作成時からある一定の期間を該オブジェクトの有効期間とすることもできる。また例えば、システムの規定値を永久に削除しないような有効期間（ $-\infty \sim +\infty$ ）とすることもできよう。これらの規定値は、システム的环境設定でユーザ（特に許されたユーザのみ）が自由に設定するようにしても良いし、ユーザが個々の規定値を設定するようにしても良い。

【0031】次に、過去に発生しているオブジェクト識別子OIDの番号に基づいて、新たなOID識別子を得る（ステップS404）。そして、そのオブジェクト識別子OIDと上記有効期間の組を有効期間表208に追

加登録する（ステップS405）。

【0032】次に、永続オブジェクトのもつオブジェクト情報を二次記憶装置105中に書き込む（ステップS406）。

【0033】最後に、永続オブジェクトの作成要求元に返値を返す（ステップS407）。この返り値は、例えば二次記憶装置105にオブジェクトを記憶するだけの容量が確保されたかどうか等、処理が正常に進んだかどうかを示す情報である。

【0034】以上の結果、実施例のデータベースにおいては、原則として個々のオブジェクトにはオブジェクト識別子OIDとその有効期間情報が作成されることになる。

【0035】次に、実施例におけるガーベジコレクション処理内容を図5のフローチャートに従って説明する。

【0036】まず、最初に、上記ガーベジコレクション機構207はガーベジになりうる永続オブジェクトを一つ選択する（ステップS501）。例えば、上記オブジェクト処理システム202が管理するオブジェクト識別子OIDから成る表（上記データベースの内部処理機構204が管理する表）を検索し、オブジェクト識別子OIDを順番に一つずつ選択してもよいし（ケース1）、上記有効期間表208から直接オブジェクト識別子OIDを順番に一つずつ選択してもよい（ケース2）。

【0037】次に、当該永続オブジェクトのオブジェクト識別子OIDから上記有効期間表208を検索し、そのオブジェクトの有効期間を読み出す（ステップS502）。

【0038】そして、当該永続オブジェクトの有効期間と、システムが保持している現在の時刻（日付も当然含まれる）を比較し、そのオブジェクトが有効期間外であるか内であるかを判断する（ステップS503）。

【0039】有効期間外であると判断した場合（YESの場合）は、該当する永続オブジェクトを消去すると共に、有効期間表208の該当箇所も削除する。

【0040】有効期間を過ぎていない場合（NOの場合）はそのまま終了する。

【0041】以上の如く、本実施例によれば、個々のオブジェクトが自身の有効期間を示す情報を持ち、個々のオブジェクトを特定する情報と有効期間の対で構成される専用の情報（テーブル）を持つことで、システムに係る負担が小さくなり、不要なオブジェクトをすばやく、且つ、簡単に探し出すことが可能になる。しかも、不用意にガーベジオブジェクトが増加することもなくなる。

【0042】尚、上記例では、有効期間表208には、オブジェクト識別子OIDとその有効期間の対を記憶させたが、オブジェクト名とその存在位置を示す情報を含

めるようにしても良い。この場合、有効期間表208の情報量は上記例と比較して多少多くなるが、オブジェクトを探し出す処理を高速にできる。

【0043】＜他の実施例の説明＞

【実施例2】オブジェクトの有効期間を図2の有効期間表208を利用するのではなく、各永続オブジェクトのデータ中に有効期間を指定するものである。オブジェクト処理システムのガーベジコレクション機構207が各永続オブジェクトから有効期間を読みだし、有効期限を過ぎたオブジェクトのみをガーベジコレクションの対象とするものである。ガーベジコレクション機構207は、最初に有効期間を調査して、有効期限内のオブジェクトはガーベジとは見なさず次のオブジェクトの調査に移動するので、上記実施例と比較して処理速度は多少劣るものの、ガーベジコレクション処理を軽減することができる。

【0044】【実施例3】図4において、オブジェクトの有効期間を指定しない場合、システムの既定値を明示的な「有効期間なし」とするものである。

【0045】例えば、「有効期間なし」の場合、有効期間表208には何も書き込まないようにもできる。この場合、有効期間表208が大きくなるのを抑制することも可能になる。さらに、ガーベジコレクション機構207がガーベジの対象として最初に有効期間表208からオブジェクト識別子OIDを選択する場合（上記「ケース2」）、「有効期間なし」のオブジェクトは自動的にガーベジの対象から除外されるので、ガーベジの対象から除外したいオブジェクトは「有効性なし」として、作成することができる。

【0046】また例えば、「有効期間なし」の場合、上記有効期間表208には「有効期間なし」を示す情報のみを書き込む様にもできる。ガーベジコレクション機構207が「有効期間なし」の永続オブジェクトをガーベジの対象とするかどうかは、ガーベジコレクション処理の時点で決めることができるようになる。この場合は、明示的にオブジェクトの有効期間を設定しなくても、ガーベジコレクション処理の時点でガーベジかどうか決まることになる。

【0047】【実施例4】図4において、オブジェクトの有効期間を指定しない場合、各クラスの既定値を利用しても良い。例えば、有効期間を指定せずにオブジェクトを作成すると作成時からある一定の期間が該オブジェクトの有効期間となる。各クラス毎に既定値を変えることができるので細かい制御が可能になる。クラスの既定値をクラス変数として可変にできる。また、クラスオブジェクトを永続オブジェクトにして二次記憶装置105に保存し、クラスオブジェクト自体の有効期間が過ぎると、該クラスのインスタンスを全てガーベジ扱いにするようにもできる。また、クラスオブジェクトの有効期間を該クラスのインスタンス全ての有効期間を含む期間

になるよう設定すると、該クラスのインスタンス全てがガーベジ扱いになると同時にクラスオブジェクトもガーベジ扱いになる。

【0048】【実施例5】クラスのバージョンを複数にしてスキーマを扱うデータベースにおいて、各クラスの差分情報及び最初のバージョンの情報のみを記憶している場合、あるバージョンのクラスオブジェクトの有効期間が過ぎた時点において該クラスオブジェクトを削除する際に、該クラスオブジェクトのバージョン以前の古いバージョンのクラスオブジェクトをマージし、スキーマの整合性を保つようにしても良い。

【0049】【実施例6】実施例5において、クラスオブジェクトの有効期間の代わりにクラスバージョン番号を使用し、バージョンがある番号を越えた時点で該バージョン以前のクラスオブジェクトをマージし、不要なクラスオブジェクト及び不要なクラスのインスタンスを消去しても良い。最大のバージョン番号は、各スキーマ毎に指定できる。指定しない場合は、システムの既定値にもできるし、各スキーマ毎の既定値にもできる。

【0050】【実施例7】実施例5においてあるクラスの有効期間が過ぎた時点tにおいて各クラスオブジェクトを削除する際に、該クラスオブジェクトのバージョンよりも古いバージョンのクラスオブジェクト全てについて有効期間を検査し全ての有効期間が前記t以前の場合のみ、マージ及び削除を行なうものである。

【0051】【実施例8】複合オブジェクトにおいて、複合オブジェクトの有効期間を指定する際に、該複合オブジェクトの要素であるオブジェクト全てについて有効期間を検査し、全ての要素の有効期間及び該複合オブジェクトの有効期間を含む有効期間Iを計算し、全ての要素の有効期間及び該複合オブジェクトの有効期間を該有効期間Iに設定しても良い。このようにして、複合オブジェクトの全ての要素の有効期間及び該複合オブジェクトの有効期間は等しくなるため、ガーベジ判定の際に、該複合オブジェクトの要素であるオブジェクト全てについて有効期間を検査する必要がなくなり、複合オブジェクトと全ての要素を同時に消去することができるようになる。

【0052】【実施例9】実施例8において、複合オブジェクトの有効期間を指定していない場合に、該複合オブジェクトの要素であるオブジェクト全てについて有効期間を検査し、全ての有効期間を含む有効期間を該複合オブジェクトの有効期間としても良い。

【0053】【実施例10】実施例8において、複合オブジェクトに要素オブジェクトを登録する際に、該要素オブジェクトの有効期間を検査し、全ての有効期間を含む有効期間を該複合オブジェクト及び該要素オブジェクトの有効期間としても良い。

【0054】【実施例11】ある一定の期間毎にデータベースの情報を二次記憶装置105に書き込むシステム

の場合、永続オブジェクトの有効期間を書き込み時点以前に指定して該オブジェクトを作成しようとしたとき、該永続オブジェクトを主記憶装置103にのみ存在する一時的オブジェクトとして作成し、二次記憶装置105および有効期間表208には書き込まないようにしても良い。このようにして無駄な書き込み処理を節約できるようになる。

【0055】【実施例12】二次記憶装置105を複数個持つシステムの場合、オブジェクトの有効期間毎に異なる二次記憶装置105に記憶し、ある期間を過ぎた時点で二次記憶装置105全ての情報を消去するようにしても良い。例えば、物理的に二次記憶装置105を初期化することもできる。また、二次記憶装置105の情報を消去せずに単に全て無意味な情報として上書き可能とすることもできる。この例では、システム全体で「二次記憶装置の識別子、有効期間」からなる表Tを用意し、二次記憶装置105毎に有効期間表208を用意し、オブジェクトを二次記憶装置105に保存する際に、該表Tから有効期間I1を読みだし、該オブジェクトの有効期間I2が該有効期間I1に含まれる場合は、該二次記憶装置に書き込み、含まれない場合は、該有効期間I1及びI2を含む有効期間I3が最小になるようなI1を計算し、対応する二次記憶装置にオブジェクトを書き込み、該表TのI1をI3に変更する。このようにして、カーページ処理機構は該表Tから有効期間を読み出し、二次記憶装置単位で消去することができるようになる。この例では、二次記憶装置単位で消去するので、有効期間表208も同時に消去される。

【0056】【実施例13】実施例12において、二次記憶装置105を複数のパーティションに分割して使用するシステムの場合二次記憶装置単位の代わりにパーティション単位で、各オブジェクトの有効期間毎に異なるパーティションに記憶しある期間を過ぎた時点で、該パーティション全ての情報を消去するものである。また、パーティションの情報を消去せずに単に全て無意味な情報として上書き可能とすることもできる。

【実施例14】図4において、永続オブジェクトの有効期間を指定する際に、最初の時刻t1を省略して、消去する時刻t2のみを指定した場合、最初の時刻t1をシステムの実時間rtに自動的に設定するものである。最初の時刻t1を省略して永続オブジェクトを作成できるので、作成処理が簡素化できる。この例において、消去する時刻t2がシステムの実時間rtよりも以前になるように指定した場合($t2 \leq rt$)には、オブジェクトの作成処理を異常終了して異常を示す返値を返してもよい(ケース3)、そのままオブジェクトの作成処理を続行し、主記憶装置103には書き込まない(一時的なオブジェクトとして作成する)ようにしてもよい(ケース4)、そのまま永続オブジェクトの作成処理を続行し、主記憶装置103にオブジェクトを作成し、有効期

間表208および二次記憶装置105に書き込み、ガーベージ処理の際にガーベージとして消去してもよい(ケース5)。

【実施例15】図4において、永続オブジェクトの有効期間を指定する際に、最初の時刻t1をシステムの実時間rtよりも以後になるように指定した場合($t1 \geq rt$)において、オブジェクトの作成処理を異常終了せずに、そのままオブジェクトの作成処理を続行するものである。システムの実時間rtよりも以後に有効になる永続オブジェクトを指定できるので、ある一定の期間にのみ有効な永続オブジェクトを作成できる。この例において、オブジェクトの作成時には、主記憶装置105にはオブジェクトを作成するが、有効期間表208および二次記憶装置105には書き込まない(一時的なオブジェクトとして作成する)ようにし、システムの実時間rtが上記の最初の時刻t1と一致した時点において、有効期間表208および二次記憶装置105に書き込むようにしてもよい(ケース6)、オブジェクトの作成時に、主記憶装置103にオブジェクトを作成し、有効期間表208および二次記憶装置105に書き込むようにしてもよい(ケース7)。ケース7において、有効期間の最初の時刻t1がシステムの実時間rtよりも以後になるように設定できるので、ガーベージコレクション機構207は、当該永続オブジェクトをガーベージと見なして消去してもよいし、有効期間が終了するまでガーベージとみなさずに消去しないようにしてもよい。

【0057】本発明は、オブジェクト指向データベースについて述べたが、他のデータベースについても考えられる。

【0058】尚、本発明は、複数の機器から構成されるシステムに適用しても1つの機器から成る装置に適用しても良い。また、本発明は、システム或は装置にプログラムを供給することによって達成される場合にも適用できることはいうまでもない。

【0059】

【発明の効果】以上説明したように本発明によれば、システムにかかる負担を軽減し、不要なオブジェクト(情報)の増加を抑制することが可能になる。

【0060】

【図面の簡単な説明】

【図1】実施例の情報処理装置の構成を示すブロック構成図である。

【図2】実施例の情報処理装置の構成概念図である。

【図3】実施例における有効期間表の内容を示す図である。

【図4】実施例における永続オブジェクトの作成及び登録の手続きの処理手順の一部を示すフローチャートである。

【図5】ガーベージコレクション機構が永続オブジェクトの中からガーベージを選択し消去する処理の内容を示

すフローチャートである。

【符号の説明】

- 101 入力装置
- 102 中央演算処理装置 (CPU)
- 103 主記憶装置 (RAM)
- 104 読み出し専用メモリ (ROM)
- 105 二次記憶装置 (データベースのデータを保存する)
- 106 出力装置
- 201 利用者プログラム (データベースに対して処理要求を行う)
- 202 オブジェクト処理システム (データベースの内部処理を行なう。二次記憶装置及び有効期間を管理し、*

* ガーベージの消去を行なう)

- 204 データベースの内部処理機構
- 205 二次記憶装置の管理機構
- 206 有効期間長の管理機構
- 207 ガーベージコレクション機構
- 208 有効期間表 (オブジェクトと有効期間の対応表)

【手続補正2】

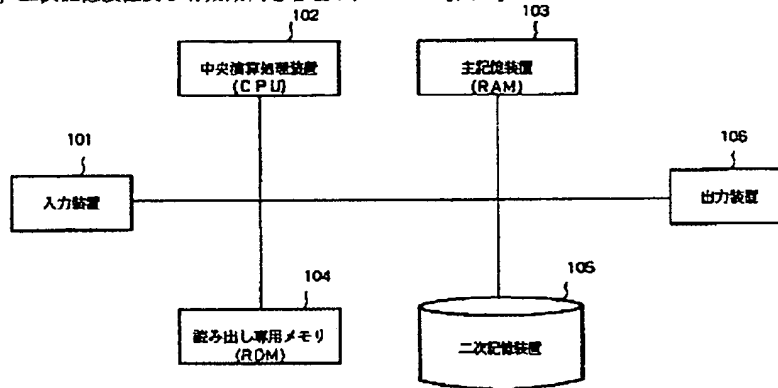
【補正対象書類名】図面

【補正対象項目名】図1

【補正方法】変更

【補正内容】

【図1】



【手続補正3】

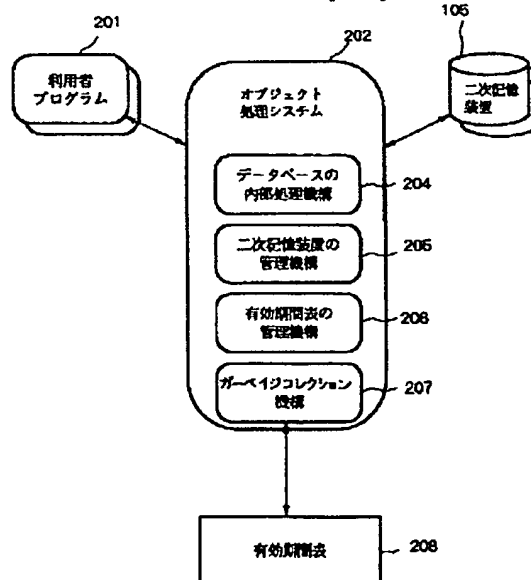
【補正対象書類名】図面

【補正対象項目名】図2

※ 【補正方法】変更

【補正内容】

※ 【図2】



【手続補正4】

【補正対象書類名】図面

【補正対象項目名】図3

【補正方法】変更

【補正内容】

* * 【図3】

オブジェクト識別子	有効期間
OID1	平元年4月17日12時30分00秒～平2年9月10日12時30分00秒
OID2	～∞ ～平2年6月30日12時30分00秒
OID3	平9年5月1日22時30分00秒～+∞
OID4	～∞～+∞

【手続補正5】

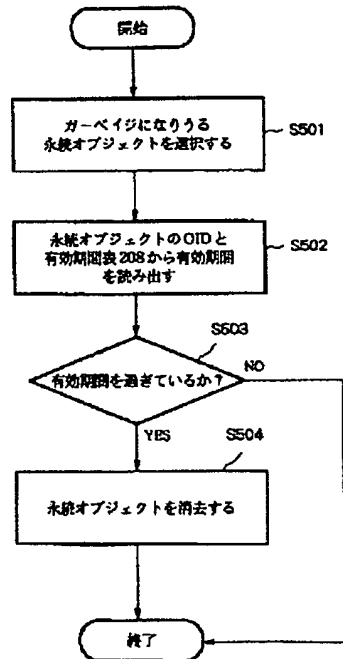
【補正対象書類名】図面

【補正対象項目名】図5

【補正方法】変更

【補正内容】

【図5】



フロントページの続き

(51) Int. Cl.⁶

識別記号

庁内整理番号

F I

技術表示箇所

G 0 6 F 17/30

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-168744

(43)Date of publication of application : 04.07.1995

(51)Int.Cl.

G06F 12/00

G06F 12/00

G06F 9/44

G06F 17/30

(21)Application number : 05-316454

(71)Applicant : CANON INC

(22)Date of filing : 16.12.1993

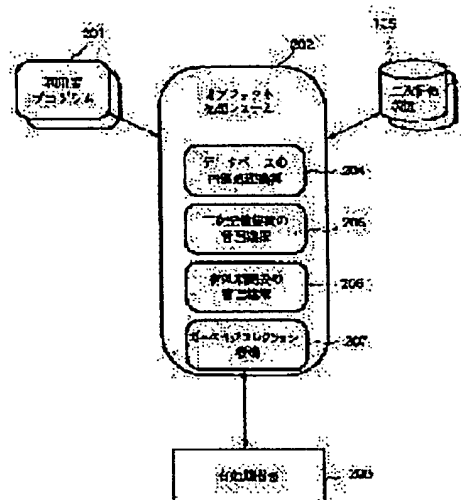
(72)Inventor : TANAKA TETSUO

(54) INFORMATION MANAGING DEVICE AND ITS CONTROL METHOD

(57)Abstract

PURPOSE: To reduce burden on a device or system and to suppress the increase of an unrequired object by storing a valid period of identification information which specifies the information of a registration object and the information of the registration object by corresponding to each other, and comparing stored information with the real time of the system, deleting registration information in an invalid period.

CONSTITUTION: When an object (information) is registered on a secondary memory device 105, the validity of the object is set. A pair of object identifier and validity is registered on a validity table 208 additionally. When the garbage processing of the object is performed, the validity of a permanent object is compared with the present time (including a date), and it is judged whether the object is within or outside the validity. When it is judged that the object is outside the validity, a corresponding permanent object is erased, and also, the corresponding part of the validity table 208 is deleted.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

* NOTICES *

JPO and NCIPJ are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] In the information management equipment which memorizes two or more information to predetermined storage, and manages it A period setting means to set up the shelf-life of the information concerned in case storage registration of the information is carried out. A storage means to match and memorize the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration. Information management equipment characterized by having a deletion means to delete registration information which compares the information and the real time of day of a system which are memorized by this storage means, and is in an invalid period.

[Claim 2] In the control approach of information management equipment of memorizing two or more information to predetermined storage, and managing it The period setting process of setting up the shelf-life of the information concerned in case storage registration of the information is carried out. The storage process which matches and memorizes the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration. The control approach of the information management equipment characterized by having the deletion process which deletes registration information which compares the information and the real time of day of a system which are memorized according to this storage process, and is in an invalid period.

[Translation done.]

* NOTICES *

JP0 and NCIP1 are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Industrial Application] This invention relates to information management equipment and its control approach, the information management equipment that performs garbage processing of the object in a database in detail, and its control approach.

[0002]

[Description of the Prior Art] Usually, the garbage collection function of an object oriented database processes it by the background or the foreground, and it discovers the permanent object (garbage) which became unnecessary, and releases the information on the permanent objects (for example, a document, an image, etc.) from mass secondary storages, such as a hard disk and a magneto-optic disk, and he is trying to prepare the processing facility of dedication in database system, and to delete it from a database management list.

[0003]

[Problem(s) to be Solved by the Invention] However, in the above-mentioned conventional example, in order to act so that the garbage collection function of an object processing system may discover garbage and may perform release and deletion, there are the following problems.

[0004] (1) The processing for a garbage collection becomes the burden of a system.

[0005] (2) In order for the permanent object which can become the element of a compound object to judge whether it is garbage, it is necessary to inspect whether the element of this compound object is the element of other compound objects.

[0006] (3) As the compound object which consists of permanent objects judges whether it is garbage, in order to guarantee that the element and configuration of a compound object are not changed, it is necessary to perform transaction processing.

[0007] (4) When a garbage collection is automatically performed by the system, a user cannot know the time amount from which garbage is eliminated.

[0008] Furthermore, since a shelf-life was not able to be specified as the object of an object processing system, there were the following faults.

[0009] (5) A shelf-life cannot be specified to a permanent object effective only in a certain fixed period.

[0010] (6) To an unnecessary permanent object, it is necessary to delete clearly before this period at a certain fixed period.

[0011] (7) A user cannot specify the time of day which makes a change to the secondary storage for creating / eliminating a permanent object, and the table of a system.

[0012]

[Means for Solving the Problem] This invention tends to be made in view of this trouble, the burden placed on equipment or a system tends to be mitigated, and it is going to offer the information management equipment which makes it possible to control the increment in an unnecessary object, and its control approach.

[0013] In order to solve this technical problem, the information management equipment of this invention is equipped with the configuration shown below. Namely, memorize two or more information to predetermined storage, and it sets to the information management equipment to manage. A period setting means to set up the shelf-life of the information concerned in case storage registration of the information is carried out. A storage means to match and memorize the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration is compared with the information and the real time of day of a system which are memorized by this storage means, and it has a deletion means to delete registration information in an invalid period.

[0014] Moreover, the control approach of the information management equipment of this invention is equipped with the process shown below. Namely, memorize two or more information to predetermined storage, and it sets to the control approach of the information management equipment to manage. The period setting process of setting up the shelf-life of the information concerned in case storage registration of the information is carried out. The storage process which matches and memorizes the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration is compared with the information and the real time of day of a system which are memorized according to this storage process, and it has the deletion process which deletes registration information in an invalid period.

[0015]

[Function] In the configuration or process of this this invention, when it is going to register a certain information, the shelf-life of the information for [the] registration is set up. And storage maintenance of the pair of a shelf-life which had the identification information for [the] registration set up is carried out. And as compared with the real time of a system, it deletes about the information which is an invalid period based on this memorized information.

[0016]

[Example] The example which starts this invention according to a drawing below is explained to a detail.

[0017] Drawing 1 is the block diagram showing the configuration of the information processor of an example.

[0018] In this drawing, the arithmetic and program control (henceforth CPU) with which 101 controls input units, such as a keyboard and a mouse, and 102 controls the whole equipment, the main storage (following, RAM) which loads the program which

uses 103 by the system, and data, and 104 are read-only memories (ROM) which have memorized a boot program, font data, etc. which are performed at the time of starting of CPU102. 105 is a secondary storage (here, it considers as a hard disk drive unit) data and the database processing program of a database are remembered to be, and this program will be loaded to RAM103 and will be performed. 106 is output units, such as a terminal and a printer.

[0019] Drawing 2 is a configuration conceptual diagram in case the database program in the above-mentioned configuration is loaded to RAM103.

[0020] In this drawing, 201 is a shelf-life table (conversion table of an object and a shelf-life) where a user's program (a processing demand is performed to a database) and 202 are carried out at an object processing system (it consists of the internal-processing device 204 of a database, a control mechanism 205 of a secondary storage, a control mechanism 206 of a shelf-life table, and a garbage collection control mechanism 207), and storage maintenance of 208 is carried out into secondary storage 105, and is mentioned later for details.

[0021] In the above-mentioned configuration, CPU102 in drawing 1 and RAM103 grade constitute the object processing system 202 and a user's program 201.

[0022] Drawing 3 shows an example of the shelf-life table 208 in drawing 2. In this drawing, "an object identifier (OID)" is the information for identifying the object in a system uniquely. Moreover, a "shelf-life" is time information which shows the shelf-life of a corresponding object, and holds the information which shows whether it is valid from when before when. That is, if it is within this period, a system guarantees existence of that object, the object outside that period regards it as a garbage object, and opening wide and deleting from secondary storage 105 is shown.

[0023] For example, the object of an object identifier "OID1" shows that 12:30 00 seconds on September 10, Heisei 2 after 30 minutes and 00 seconds are a shelf-life, and the information on this object is saved in a secondary storage 203 in this shelf-life at Narimoto Taira year 12:00 on April 17.

[0024] Moreover, it is shown that storage maintenance is carried out into a secondary storage 203 unless a shelf-life is effective and eliminates clearly in all periods, and the object of an object identifier "OID4" is not set as the object of a garbage collection.

[0025] In addition, about OID2 and OID3, I will guess easily from the above-mentioned explanation.

[0026] The information on the shelf-life table 208 of drawing 2 may be saved apart from the data of a permanent object at the secondary storage 203 of drawing 2, and may save the information on front [whole] as one certain permanent object at a secondary storage 203. Moreover, you may save for every permanent object as information in each permanent object at a secondary storage 203.

[0027] Moreover, you may make it the information on the shelf-life table 208 specify only the time of day which omits and eliminates the first time of day. In this case, the time of day which writes in the information on a new object into a secondary storage 203 (registration) cannot be specified.

[0028] Now, in the above-mentioned configuration, although the well-known additional information (information on a keyword, an implementer, or others) over an object is attached in case new data (object) are registered into a system, at this time, a user directs the input of the shelf-life to that object (if), and directs a registration request to a system.

[0029] Drawing 4 shows the procedure of the system which received this registration request.

[0030] It confirms whether specify the shelf-life of an object that the creation demand of a permanent object occurs (decision step S402). (step S401) When the shelf-life of an object is not specified, the default value of a system is used (step S403). (when it is NO) For example, a certain fixed period can also be made into the shelf-life of this object for the default value of a system from creation time. Moreover, for example, it can also consider as a shelf-life (- infinity+infinity) which does not delete the default value of a system eternally. A user (only user allowed especially) may be made to set up such default value freely by the configuration of a system, and a user may be made to set up each default value.

[0031] Next, a new OID identifier is obtained based on the number of the object identifier OID generated in the past (step S404). And additional registration of the group of the object identifier OID and above-mentioned shelf-life is carried out in the shelf-life table 208 (step S405).

[0032] Next, the object information which a permanent object has is written in into a secondary storage 203 (step S406).

[0033] Finally, a ** value is returned to the creation demand origin of a permanent object (step S407). This return value is information which shows whether whether only the capacity which memorizes an object having been secured to secondary storage 105, and processings progressed normally.

[0034] In the database of an example, OID and its shelf-life information will be created by each object in principle the above result.

[0035] Next, the contents of garbage collection processing in an example are explained according to the flow chart of drawing 5.

[0036] First, the above-mentioned garbage collection device 207 chooses as the beginning one permanent object which can become garbage (step S501). For example, the table (table which the internal-processing device 204 of the above-mentioned database manages) which consists of the object identifier OID which the above-mentioned object processing system 202 manages may be searched, every one object identifier OID may be chosen in order, and every one direct object identifier OID may be chosen from the above-mentioned shelf-life table 208 in order (case 2). (case 1)

[0037] Next, the above-mentioned shelf-life table 208 is searched from the object identifier OID of the permanent object concerned, and the shelf-life of the object is read (step S502).

[0038] And the shelf-life of the permanent object concerned is compared with the current time of day (naturally the date is also contained) which the system holds, and it judges whether the object is outside a shelf-life, or it is inside (step S503).

[0039] When it is judged that it is outside a shelf-life (in the case of YES), while eliminating the corresponding permanent object, the applicable part of the shelf-life table 208 is also deleted.

[0040] When having not passed over the shelf-life (in the case of NO), it ends as it is.

[0041] Like the above, it has the information each object indicates an own shelf-life to be, the burden concerning a system becomes small according to this example, by having the information which specifies each object, and the information on the dedication which consists of pairs of a shelf-life (file), and it becomes possible to discover an unnecessary object simply quickly.

And it becomes, without a garbage object increasing carelessly.

[0042] In addition, although the shelf-life table 208 was made to memorize the pair of Object ID and its shelf-life, you may make it include the information which shows an object name and its existence location in it in the above-mentioned example. In this case, although the amount of information of the shelf-life table 208 increases somewhat as compared with the above-mentioned example, processing which discovers an object is made to a high speed.

[0043] The shelf-life table 208 of drawing 2 is not used for the shelf-life of the explanation > [example 2] object of an example besides <, but a shelf-life is specified in the data of each permanent object. The garbage collection device 207 of an object processing system reads a shelf-life from each permanent object, and sets only the object which passed over the expiration date as the object of a garbage collection. The garbage collection device 207 can mitigate garbage collection processing, although some processing speed is inferior as compared with the above-mentioned example, since a shelf-life is investigated first, and the object within an expiration date does not regard it as garbage but it moves to investigation of the following object.

[0044] In [example 3] drawing 4, when not specifying the shelf-life of an object, the default of a system is made "with no shelf-life". [explicit]

[0045] For example, in "having no shelf-life", nothing can be written in the shelf-life table 208. In this case, it also becomes possible to control that the shelf-life table 208 becomes large. Furthermore, since the object of "having no shelf-life" is automatically excepted from the object of garbage when the garbage collection device 207 chooses the object identifier OID from the shelf-life table 208 first as an object of garbage (above "a case 2"), an object to except from the object of garbage can be created as "with no effectiveness."

[0046] Moreover, for example, in "having no shelf-life", only the information which shows "he has no shelf-life" can be written in the above-mentioned shelf-life table 208. It can be decided now at the time of garbage collection processing whether the garbage collection device 207 sets the permanent object of "having no shelf-life" as the object of garbage. In this case, even if it does not set up the shelf-life of an object clearly, it will be decided at the time of garbage collection processing whether it is garbage.

[0047] In [example 4] drawing 4, when not specifying the shelf-life of an object, the default of each class may be used. For example, if an object is created without specifying a shelf-life, a certain fixed period will turn into a shelf-life of this object from creation time. Since a default is changeable for every class, fine control is attained. The default of a class is made to adjustable as a class variable. Moreover, if a class object is made into a permanent object, it saves at a secondary storage 203 and the shelf-life of the class object itself passes, all the instances of this class can be treated as garbage. Moreover, if the shelf-life of a class object is set up so that the period containing the shelf-lives of all the instances of this class may come, a class object will also become garbage treatment while all the instances of this class become garbage treatment.

[0048] the database which makes the version of a [example 5] class plurality and treats a schema — setting — the difference of each class — in case this class object is deleted when only information and the information on the first version are memorized, and the shelf-life of the class object of a certain version passes, the class object of the old version of the version or earlier of this class object is merged, and you may make it maintain the adjustment of a schema

[0049] "In the example 6] example 5, a class version number may be used instead of the shelf-life of a class object, when a number with a version is exceeded, the class object before this version may be merged, and the instance of an unnecessary class object and an unnecessary class may be eliminated. The greatest version number can be specified for every schema. When not specifying, it is made also to the default of a system and is made also to the default for every schema.

[0050] In case each class object is deleted in t the time of the shelf-life of the class set in the "example 7" example 5 passing, a shelf-life is inspected about all the class objects of a version older than the version of this class object, and only when all shelf-lives are before t, merge and deletion are performed.

[0051] In a "example 8" compound object, in case the shelf-life of a compound object is specified, a shelf-life may be inspected about all the objects that are the elements of this compound object, the shelf-life I containing the shelf-life of all elements and the shelf-life of this compound object may be calculated, and the shelf-life of all elements and the shelf-life of this compound object may be set as this shelf-life I. Thus, since the shelf-life of all the elements of a compound object and the shelf-life of this compound object become equal, in the case of a garbage judging, it becomes unnecessary to inspect a shelf-life about all the objects that are the elements of this compound object, and they can eliminate a compound object and all elements to coincidence.

[0052] In the [example 9] example 8, when the shelf-life of a compound object is not specified, it is good also considering the shelf-life which inspects a shelf-life about all the objects that are the elements of this compound object, and contains all shelf-lives as a shelf-life of this compound object.

[0053] In the [example 10] example 8, in case an element object is registered into a compound object, it is good also considering the shelf-life which inspects the shelf-life of this element object and contains all shelf-lives as a shelf-life of this compound object and this element object.

[0054] [Example 11] When in the case of the system which writes the information on a database in a secondary storage 203 for every fixed period of a certain the shelf-life of a permanent object tends to be written in, it is going to specify before a time and it is going to create this object, this permanent object is created as a temporary object which exists only in main storage 103, and you may make it not write in a secondary storage 203 and the shelf-life table 208. Thus, useless write-in processing can be saved now.

[0055] the time of memorizing to a different secondary storage 203 for every shelf-life of an object in the case of a system with two or more [example 12] secondary storages 203, and passing over a certain period — a secondary storage 203 — you may make it eliminate all information For example, a secondary storage 203 can also be initialized physically. Moreover, overwrite can also only be altogether made possible as meaningless information, without eliminating the information on a secondary storage 203. In this example, T which consists of "an identifier of a secondary storage and a shelf-life" by the whole system is prepared. In case the shelf-life table 208 is prepared every secondary storage 203 and an object is saved at a secondary storage 203 When a shelf-life I1 is read from this table T and the shelf-life I2 of this object is contained at this shelf-life I1 It writes in this secondary storage, and when not contained, I1 from which the shelf-life I3 containing these shelf-lives I1 and I2 becomes min is calculated, an object is written in a corresponding secondary storage, and T11 is changed into this table I3. Thus, a KABEJI

processor can read a shelf-life and can eliminate it now from this table T per secondary storage. In this example, since it eliminates per secondary storage, the shelf-life table 208 is also eliminated by coincidence.

[0056] In the [example 13] example 12, in the case of the system which divides and uses a secondary storage 203 for two or more partitions, it is a partition unit instead of a secondary-storage unit, and when it memorizes to a different partition for every shelf-life of each object and passes over a certain period, the information on these all partitions is eliminated. Moreover, overwrite can also only be altogether made possible as meaningless information, without eliminating the information on a partition.

[0057] Other databases are considered although this invention described the object oriented database.

[0058] In addition, even if it applies this invention to the system which consists of two or more devices, it may be applied to the equipment which consists of one device. Moreover, it cannot be overemphasized that this invention can be applied also when attained by supplying a program to a system or equipment.

[0059]

[Effect of the Invention] As explained above, the burden which is placed on a system according to this invention is mitigated, and it becomes possible to control the increment in an unnecessary object (information).

[0060]

[Translation done.]

* NOTICES *

JPO and NCIPJ are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the block block diagram showing the configuration of the information processor of an example.

[Drawing 2] It is the configuration conceptual diagram of the information processor of an example.

[Drawing 3] It is drawing showing the contents of the shelf-life table in an example.

[Drawing 4] It is the flow chart which shows a part of procedure of creation of the permanent object in an example, and the procedure of registration.

[Drawing 5] It is the flow chart which shows the contents of the processing to which a garbage collection device chooses and eliminates garbage out of a permanent object.

[Description of Notations]

1 User Program (Processing Demand is Performed to Database)

2 Object processing system (internal processing of a database is performed.) A secondary storage and a shelf-life are managed and garbage is eliminated.

3 Secondary Storage (Data of Database are Saved)

4 Shelf-life Table (Conversion Table of Object and Shelf-life)

[Translation done.]

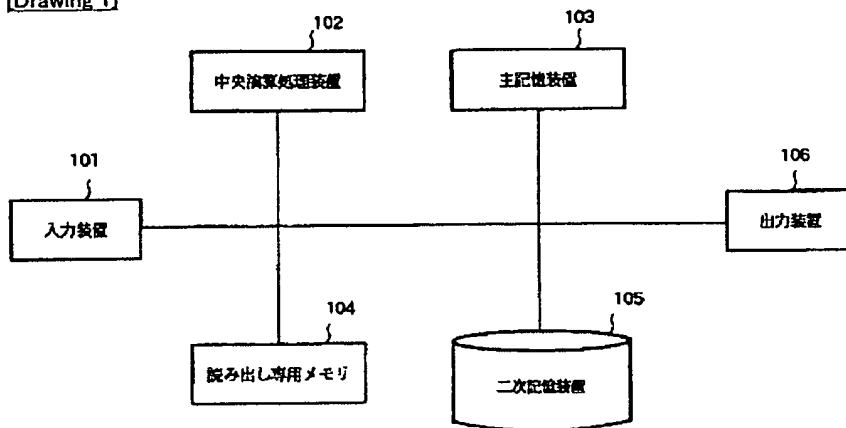
* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

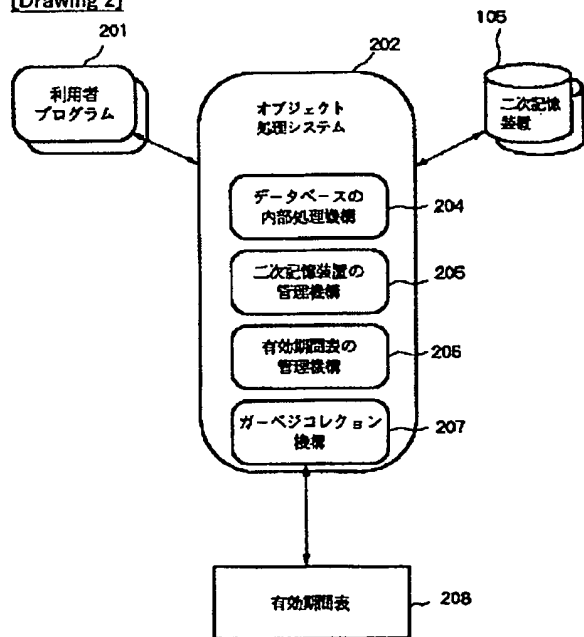
- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DRAWINGS

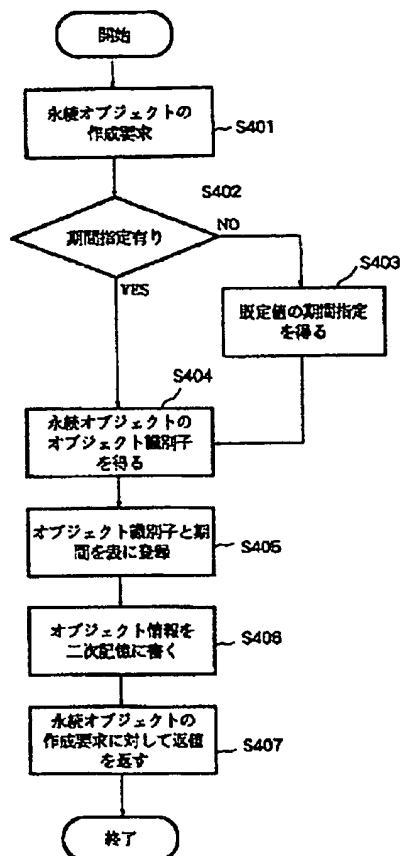
[Drawing 1]



[Drawing 2]



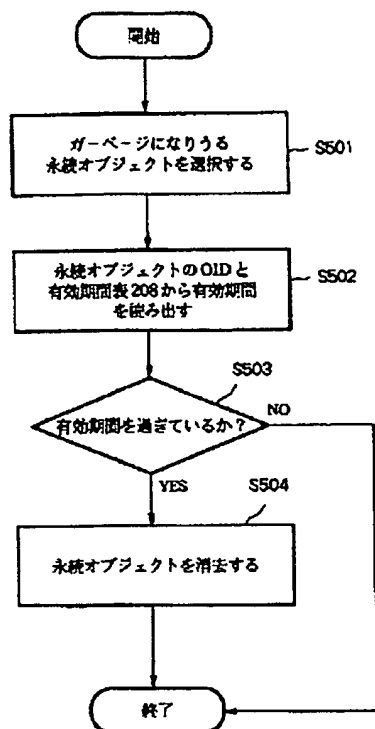
[Drawing 4]



[Drawing 3]

オブジェクト識別子	有効期間
OID1	平1年5月1日12時30分00秒～平3年7月1日12時30分00秒
OID2	－∞～平2年6月30日12時30分00秒
OID3	平3年5月1日22時30分00秒～＋∞
OID4	－∞～＋∞

[Drawing 5]



[Translation done.]

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

WRITTEN AMENDMENT

[a procedure revision]

[Filing Date] May 9, Heisei 6

[Procedure amendment 1]

[Document to be Amended] Specification

[Item(s) to be Amended] Whole sentence

[Method of Amendment] Modification

[Proposed Amendment]

[Document Name] Specification

[Title of the Invention] Information management equipment and its control approach

[Claim(s)]

[Claim 1] In the information management equipment which memorizes two or more information to predetermined storage, and manages it,

A period setting means to set up the shelf-life of the information concerned in case storage registration of the information is carried out,

A storage means to match and memorize the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration,

Information management equipment characterized by having a deletion means to delete registration information which compares the information and the real time of day of a system which are memorized by this storage means, and is in an invalid period.

[Claim 2] In the control approach of information management equipment of memorizing two or more information to predetermined storage, and managing it,

The period setting process of setting up the shelf-life of the information concerned in case storage registration of the information is carried out,

The storage process which matches and memorizes the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration,

The control approach of the information management equipment characterized by having the deletion process which deletes registration information which compares the information and the real time of day of a system which are memorized according to this storage process, and is in an invalid period.

[Detailed Description of the Invention]

[0001]

[Industrial Application] This invention relates to information management equipment and its control approach, the information management equipment that performs garbage processing of the object in a database in detail, and its control approach.

[0002]

[Description of the Prior Art] Usually, the garbage collection function of an object oriented database processes it by the background or the foreground, and it discovers the permanent object (garbage) which became unnecessary, and releases the information on the permanent objects (for example, a document, an image, etc.) from mass secondary storages, such as a hard disk and a magneto-optic disk, and he is trying to prepare the processing facility of dedication in database system, and to delete it from a database management list.

[0003]

[Problem(s) to be Solved by the Invention] However, in the above-mentioned conventional example, in order to act so that the garbage collection function of an object processing system may discover garbage and may perform release and deletion, there are the following problems.

[0004] (1) The processing for a garbage collection becomes the burden of a system.

[0005] (2) In order for the permanent object which can become the element of a compound object to judge whether it is garbage, it is necessary to inspect whether the element of this compound object is the element of other compound objects.

[0006] (3) As the compound object which consists of permanent objects judges whether it is garbage, in order to guarantee that the element and configuration of a compound object are not changed, it is necessary to perform transaction processing.

[0007] (4) When a garbage collection is automatically performed by the system, a user cannot know the time amount from which garbage is eliminated.

[0008] Furthermore, since a shelf-life was not able to be specified as the object of an object processing system, there were the following faults.

[0009] (5) A shelf-life cannot be specified to a permanent object effective only in a certain fixed period.

[0010] (6) To an unnecessary permanent object, it is necessary to delete clearly before this period at a certain fixed period.

[0011] (7) A user cannot specify the time of day which makes a change to the secondary storage for creating / eliminating a permanent object, and the table of a system.

[0012]

[Means for Solving the Problem] This invention tends to be made in view of this trouble, the burden placed on equipment or a system tends to be mitigated, and it is going to offer the information management equipment which makes it possible to control the increment in an unnecessary object, and its control approach.

[0013] In order to solve this technical problem, the information management equipment of this invention is equipped with the configuration shown below. Namely, memorize two or more information to predetermined storage, and it sets to the information management equipment to manage. A period setting means to set up the shelf-life of the information concerned in case storage registration of the information is carried out. A storage means to match and memorize the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration is compared with the information and the real time of day of a system which are memorized by this storage means, and it has a deletion means to delete registration information in an invalid period.

[0014] Moreover, the control approach of the information management equipment of this invention is equipped with the process shown below. Namely, memorize two or more information to predetermined storage, and it sets to the control approach of the information management equipment to manage. The period setting process of setting up the shelf-life of the information concerned in case storage registration of the information is carried out. The storage process which matches and memorizes the shelf-life of the identification information which specifies the information for registration, and the information for [concerned] registration is compared with the information and the real time of day of a system which are memorized according to this storage process, and it has the deletion process which deletes registration information in an invalid period.

[0015]

[Function] In the configuration or process of this invention, when it is going to register a certain information, the shelf-life of the information for [the] registration is set up. And storage maintenance of the pair of a shelf-life which had the identification information for [the] registration set up is carried out. And as compared with the real time of a system, it deletes about the information which is an invalid period based on this memorized information.

[0016]

[Example] The example which starts this invention according to a drawing below is explained to a detail.

[0017] Drawing 1 is the block diagram showing the configuration of the information processor of an example.

[0018] In this drawing, the arithmetic and program control (henceforth CPU) with which 101 controls input units, such as a keyboard and a mouse, and 102 controls the whole equipment, the main storage (following, RAM) which loads the program which uses 103 by the system, and data, and 104 are read-only memories (ROM) which have memorized a boot program, font data, etc. which are performed at the time of starting of CPU102. 105 is secondary storages (for example, a hard disk drive unit, optical-magnetic disc equipment, etc.) data and the database processing program of a database are remembered to be, and this program will be loaded to RAM103 and will be performed. 106 is output units, such as a terminal and a printer.

[0019] Drawing 2 is a configuration conceptual diagram in case the database program in the above-mentioned configuration is loaded to RAM103.

[0020] In this drawing, 201 is a shelf-life table (conversion table of an object and a shelf-life) where a user's program (a processing demand is performed to a database) and 202 are carried out at an object processing system (it consists of the internal-processing device 204 of a database, a control mechanism 205 of a secondary storage, a control mechanism 206 of a shelf-life table, and a garbage collection device 207), and storage maintenance of 208 is carried out into the secondary storage 105, and is mentioned later for details.

[0021] In the above-mentioned configuration, CPU102 in drawing 1 and RAM103 grade constitute the object processing system 202 and a user's program 201.

[0022] Drawing 3 shows an example of the shelf-life table 208 in drawing 2. In this drawing, "an object identifier (OID)" is the information for identifying the object in a system uniquely. Moreover, a "shelf-life" is time information which shows the shelf-life of a corresponding object, and holds the information which shows whether it is valid from when before when. That is, if it is within this period, a system guarantees existence of that object, the object outside that period regards it as a garbage object, and opening wide and deleting from a secondary storage 105 is shown.

[0023] For example, the object of an object identifier "OID1" shows that 12:30 00 seconds on September 10, Heisei 2 after 30 minutes and 00 seconds are a shelf-life, and the information on this object is saved in a secondary storage 105 in this shelf-life at Narimoto Taira year 12:00 on April 17.

[0024] Moreover, it is shown that storage maintenance is carried out into a secondary storage 105 unless a shelf-life is effective and eliminates clearly in all periods, and the object of an object identifier "OID4" is not set as the object of a garbage collection.

[0025] In addition, about OID2 and OID3, I will guess easily from the above-mentioned explanation.

[0026] The information on the shelf-life table 208 of drawing 2 may be saved apart from the data of a permanent object at the secondary storage 105 of drawing 2, and may save the information on front [whole] as one certain permanent object at a secondary storage 105. Moreover, you may save for every permanent object as information in each permanent object at a secondary storage 105.

[0027] Moreover, you may make it the information on the shelf-life table 208 specify only the time of day which omits and eliminates the first time of day. In this case, the time of day which writes in the information on a new object into a secondary storage 105 (registration) cannot be specified.

[0028] Now, in the above-mentioned configuration, although the well-known additional information (information on a keyword, an implementer, or others) over an object is attached in case new data (object) are registered into a system, at this time, a user directs the input of the shelf-life to that object (if), and directs a registration request to a system.

[0029] Drawing 4 shows the procedure of the system which received this registration request.

[0030] It confirms whether the shelf-life of an object that the creation demand of a permanent object occurs (decision step S402). (step S401) When the shelf-life of an object is not specified, the default value of a system is used (step S403). (when it is NO) For example, a certain fixed period can also be made into the shelf-life of this object for the default value of a system from creation time. Moreover, for example, it can also consider as a shelf-life (-infinity--infinity) which does not delete

the default value of a system eternally. A user (only user allowed especially) may be made to set up such default value freely by the configuration of a system, and a user may be made to set up each default value.

[0031] Next, a new OID identifier is obtained based on the number of the object identifier OID generated in the past (step S404). And additional registration of the group of the object identifier OID and above-mentioned shelf-life is carried out in the shelf-life table 208 (step S405).

[0032] Next, the object information which a permanent object has is written in into a secondary storage 105 (step S406).

[0033] Finally, a ** value is returned to the creation demand origin of a permanent object (step S407). This return value is information which shows whether only the capacity which memorizes an object having been secured to the secondary storage 105, and processings progressed normally.

[0034] In the database of an example, the object identifier OID and its shelf-life information will be created by each object in principle the above result.

[0035] Next, the contents of garbage collection processing in an example are explained according to the flow chart of drawing 5.

[0036] First, the above-mentioned garbage collection device 207 chooses as the beginning one permanent object which can become garbage (step S501). For example, the table (table which the internal-processing device 204 of the above-mentioned database manages) which consists of the object identifier OID which the above-mentioned object processing system 202 manages may be searched, every one object identifier OID may be chosen in order, and every one direct object identifier OID may be chosen from the above-mentioned shelf-life table 208 in order (case 2). (case 1)

[0037] Next, the above-mentioned shelf-life table 208 is searched from the object identifier OID of the permanent object concerned, and the shelf-life of the object is read (step S502).

[0038] And the shelf-life of the permanent object concerned is compared with the current time of day (naturally the date is also contained) which the system holds, and it judges whether the object is outside a shelf-life, or it is inside (step S503).

[0039] When it is judged that it is outside a shelf-life (in the case of YES), while eliminating the corresponding permanent object, the applicable part of the shelf-life table 208 is also deleted.

[0040] When having not passed over the shelf-life (in the case of NO), it ends as it is.

[0041] Like the above, it has the information each object indicates an own shelf-life to be, the burden concerning a system becomes small according to this example, by having the information which specifies each object, and the information on the dedication which consists of pairs of a shelf-life (table), and it becomes possible to discover an unnecessary object simply quickly. And it becomes, without a garbage object increasing carelessly.

[0042] In addition, although the shelf-life table 208 was made to memorize the pair of the object identifier OID and its shelf-life, you may make it include the information which shows an object name and its existence location in it in the above-mentioned example. In this case, although the amount of information of the shelf-life table 208 increases somewhat as compared with the above-mentioned example, processing which discovers an object is made to a high speed.

[0043] Explanation [of an example besides <] >

The shelf-life table 208 of drawing 2 is not used for the shelf-life of a [example 2] object, but a shelf-life is specified in the data of each permanent object. The garbage collection device 207 of an object processing system reads a shelf-life from each permanent object, and sets only the object which passed over the expiration date as the object of a garbage collection. The garbage collection device 207 can mitigate garbage collection processing, although some processing speed is inferior as compared with the above-mentioned example, since a shelf-life is investigated first, and the object within an expiration date does not regard it as garbage but it moves to investigation of the following object.

[0044] In [example 3] drawing 4, when not specifying the shelf-life of an object, the default of a system is made "with no shelf-life". [explicit]

[0045] For example, in "having no shelf-life", nothing can be written in the shelf-life table 208. In this case, it also becomes possible to control that the shelf-life table 208 becomes large. Furthermore, since the object of "having no shelf-life" is automatically excepted from the object of garbage when the garbage collection device 207 chooses the object identifier OID from the shelf-life table 208 first as an object of garbage (above "a case 2"), an object to except from the object of garbage can be created as "with no effectiveness."

[0046] Moreover, for example, in "having no shelf-life", only the information which shows "he has no shelf-life" can be written in the above-mentioned shelf-life table 208. It can be decided now at the time of garbage collection processing whether the garbage collection device 207 sets the permanent object of "having no shelf-life" as the object of garbage. In this case, even if it does not set up the shelf-life of an object clearly, it will be decided at the time of garbage collection processing whether it is garbage.

[0047] In [example 4] drawing 4, when not specifying the shelf-life of an object, the default of each class may be used. For example, if an object is created without specifying a shelf-life, a certain fixed period will turn into a shelf-life of this object from creation time. Since a default is changeable for every class, fine control is attained. The default of a class is made to adjustable as a class variable. Moreover, if a class object is made into a permanent object, it saves at a secondary storage 105 and the shelf-life of the class object itself passes, all the instances of this class can be treated as garbage. Moreover, if the shelf-life of a class object is set up so that the period containing the shelf-lives of all the instances of this class may come, a class object will also become garbage treatment while all the instances of this class become garbage treatment.

[0048] the database which makes the version of a [example 5] class plurality and treats a schema — setting — the difference of each class — in case this class object is deleted when only information and the information on the first version are memorized, and the shelf-life of the class object of a certain version passes, the class object of the old version of the version or earlier of this class object is merged, and you may make it maintain the adjustment of a schema

[0049] "In the example 6] example 5, a class version number may be used instead of the shelf-life of a class object, when a number with a version is exceeded, the class object before this version may be merged, and the instance of an unnecessary class object and an unnecessary class may be eliminated. The greatest version number can be specified for every schema. When not specifying, it is made also to the default of a system and is made also to the default for every schema.

[0050] In case each class object is deleted in the time of the shelf-life of the class set in the "example 7" example 5 passing,

a shelf-life is inspected about all the class objects of a version older than the version of this class object, and only when all shelf-lives are before said t, merge and deletion are performed.

[0051] In a "example 8" compound object, in case the shelf-life of a compound object is specified, a shelf-life may be inspected about all the objects that are the elements of this compound object, the shelf-life I containing the shelf-life of all elements and the shelf-life of this compound object may be calculated, and the shelf-life of all elements and the shelf-life of this compound object may be set as this shelf-life I. Thus, since the shelf-life of all the elements of a compound object and the shelf-life of this compound object become equal, in the case of a garbage judging, it becomes unnecessary to inspect a shelf-life about all the objects that are the elements of this compound object, and they can eliminate a compound object and all elements to coincidence.

[0052] In the [example 9] example 8, when the shelf-life of a compound object is not specified, it is good also considering the shelf-life which inspects a shelf-life about all the objects that are the elements of this compound object, and contains all shelf-lives as a shelf-life of this compound object.

[0053] In the [example 10] example 8, in case an element object is registered into a compound object, it is good also considering the shelf-life which inspects the shelf-life of this element object and contains all shelf-lives as a shelf-life of this compound object and this element object.

[0054] [Example 11] When in the case of the system which writes the information on a database in a secondary storage 105 for every fixed period of a certain the shelf-life of a permanent object tends to be written in, it is going to specify before a time and it is going to create this object, this permanent object is created as a temporary object which exists only in main storage 103, and you may make it not write in a secondary storage 105 and the shelf-life table 208. Thus, useless write-in processing can be saved now.

[0055] the time of memorizing to a different secondary storage 105 for every shelf-life of an object in the case of a system with two or more [example 12] secondary storages 105, and passing over a certain period — a secondary storage 105 — you may make it eliminate all information. For example, a secondary storage 105 can also be initialized physically. Moreover, overwrite can also only be altogether made possible as meaningless information, without eliminating the information on a secondary storage 105. In this example, Table T which consists of "an identifier of a secondary storage and a shelf-life" by the whole system is prepared. In case the shelf-life table 208 is prepared every secondary storage 105 and an object is saved at a secondary storage 105. When a shelf-life I1 is read from this table T and the shelf-life I2 of this object is contained at this shelf-life I1. It writes in this secondary storage, and when not contained, I1 from which the shelf-life I3 containing these shelf-lives I1 and I2 becomes min is calculated, an object is written in a corresponding secondary storage, and T11 is changed into this table I3. Thus, a car BEIJI processor can read a shelf-life and can eliminate it now from this table T per secondary storage. In this example, since it eliminates per secondary storage, the shelf-life table 208 is also eliminated by coincidence.

[0056] In the [example 13] example 12, in the case of the system which divides and uses a secondary storage 105 for two or more partitions, it is a partition unit instead of a secondary-storage unit, and when it memorizes to a different partition for every shelf-life of each object and passes over a certain period, the information on these all partitions is eliminated. Moreover, overwrite can also only be altogether made possible as meaningless information, without eliminating the information on a partition.

In [example 14] drawing 4, in case the shelf-life of a permanent object is specified, the first time of day t1 is omitted, and when only the time of day t2 to eliminate is specified, the first time of day t1 is automatically set as the real time rt of a system. Since the first time of day t1 is omitted and a permanent object can be created, creation processing can be simplified. When it specifies that the time of day t2 to eliminate becomes before rather than the real time rt of a system in this example ($t2 < rt$) May return the ** value which terminates creation processing of an object abnormally and shows abnormalities, and (case 3) Continue creation processing of an object then and do not write in main storage 103, even if like (it creates as a temporary object), it is good, and (case 4) Creation processing of a permanent object may be then continued, an object may be created to main storage 103, it may write in the shelf-life table 208 and a secondary storage 105, and you may eliminate as garbage in the case of garbage processing (case 5).

In [example 15] drawing 4, in case the shelf-life of a permanent object is specified, when the first time of day t1 is specified that henceforth comes rather than the real time rt of a system ($t1 > rt$), creation processing of an object is continued as it is, without terminating creation processing of an object abnormally. Since the permanent object which becomes henceforth more effective than the real time rt of a system can be specified, a permanent object effective only in a certain fixed period can be created. In this example, although an object is created to the creation time of an object at main storage 105 [when / at which it does not write in the shelf-life table 208 and a secondary storage 105 / it was made like (it creates as a temporary object) and the real time rt of a system was in agreement with the time of day t1 of the above-mentioned beginning] It may be made to write in the shelf-life table 208 and a secondary storage 105, and an object is created to the creation time of an object at main storage 103 (case 6), and you may make it write in the shelf-life table 208 and a secondary storage 105 (case 7). It may consider that the permanent object concerned is garbage, and the garbage collection device 207 may eliminate it, and you may make it not eliminate it in a case 7, without regarding it as garbage, since it can set up so that the time of day t1 of the beginning of a shelf-life may become henceforth rather than the real time rt of a system until a shelf-life expires.

[0057] Other databases are considered although this invention described the object oriented database.

[0058] In addition, even if it applies this invention to the system which consists of two or more devices, it may be applied to the equipment which consists of one device. Moreover, it cannot be overemphasized that this invention can be applied also when attained by supplying a program to a system or equipment.

[0059]

[Effect of the Invention] As explained above, the burden which is placed on a system according to this invention is mitigated, and it becomes possible to control the increment in an unnecessary object (information).

[0060]

[Brief Description of the Drawings]

[Drawing 1] It is the block block diagram showing the configuration of the information processor of an example.

[Drawing 2] It is the configuration conceptual diagram of the information processor of an example.

[Drawing 3] It is drawing showing the contents of the shelf-life table in an example.

[Drawing 4] It is the flow chart which shows a part of procedure of creation of the permanent object in an example, and the procedure of registration.

[Drawing 5] It is the flow chart which shows the contents of the processing to which a garbage collection device chooses and eliminates garbage out of a permanent object.

[Description of Notations]

101 Input Unit

102 Arithmetic and Program Control (CPU)

103 Main Storage (RAM)

104 Read-only Memory (ROM)

105 Secondary Storage (Data of Database are Saved)

106 Output Unit

201 User Program (Processing Demand is Performed to Database)

202 Object processing system (internal processing of a database is performed.) A secondary storage and a shelf-life are managed and garbage is eliminated.

204 Internal-Processing Device of Database

205 Control Mechanism of Secondary Storage

206 Control Mechanism of Shelf-life Length

207 Garbage Collection Device

208 Shelf-life Table (Conversion Table of Object and Shelf-life)

[Procedure amendment 2]

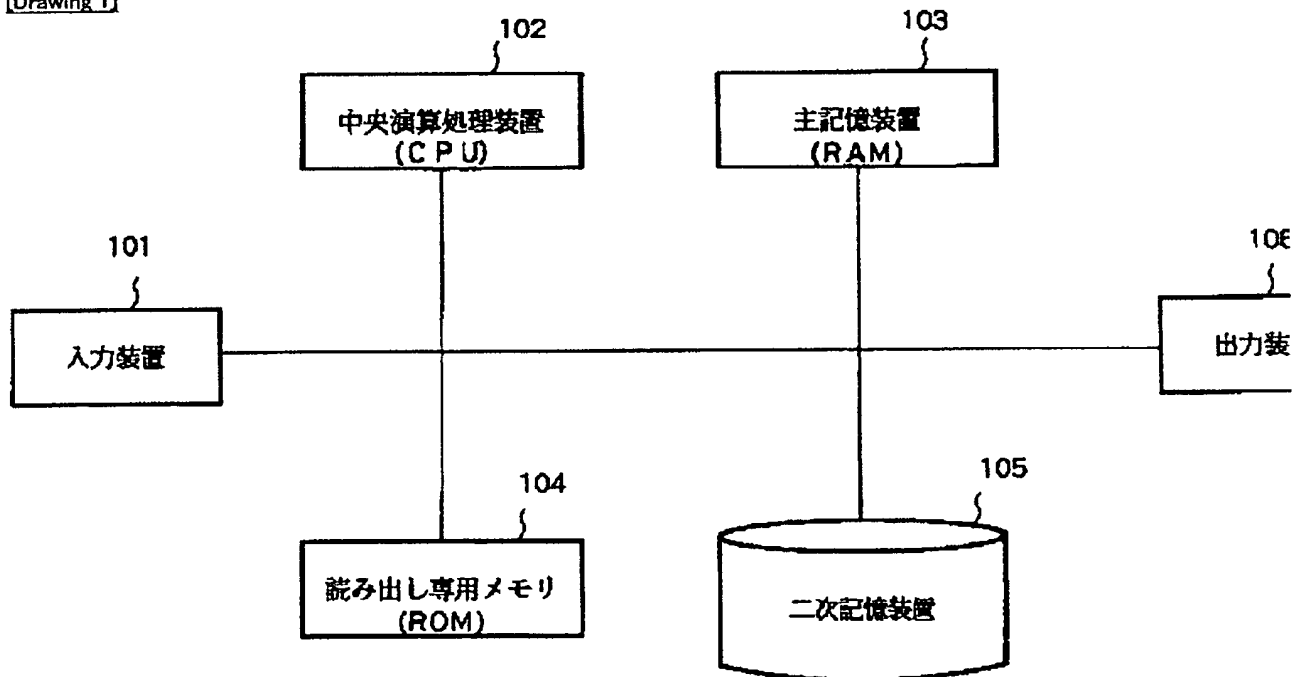
[Document to be Amended] DRAWINGS

[Item(s) to be Amended] drawing 1

[Method of Amendment] Modification

[Proposed Amendment]

[Drawing 1]



[Procedure amendment 3]

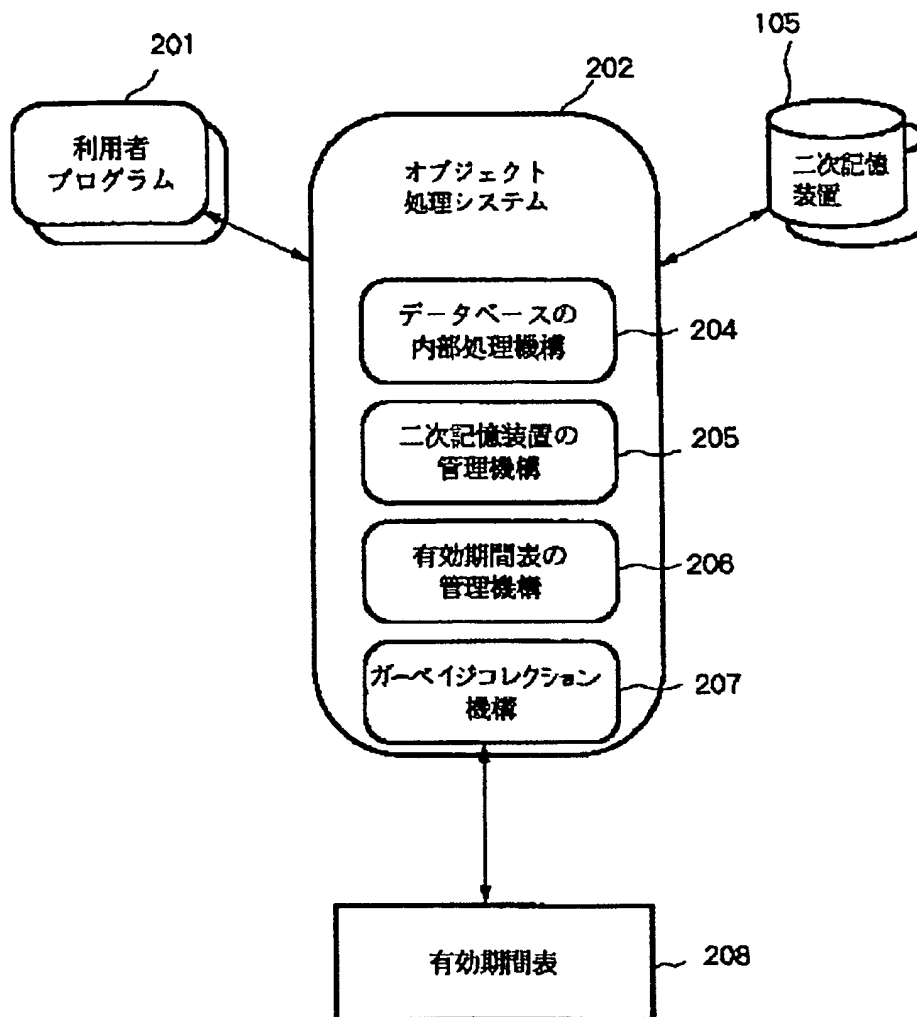
[Document to be Amended] DRAWINGS

[Item(s) to be Amended] drawing 2

[Method of Amendment] Modification

[Proposed Amendment]

[Drawing 2]



[Procedure amendment 4]

[Document to be Amended] DRAWINGS

[Item(s) to be Amended] drawing 3

[Method of Amendment] Modification

[Proposed Amendment]

[Drawing 3]

オブジェクト識別子	有効期間
OID1	平元年4月17日12時30分00秒～平2年9月10日12時30分00秒
OID2	-∞～平2年6月30日12時30分00秒
OID3	平3年5月1日22時30分00秒～+∞
OID4	-∞～+∞

[Procedure amendment 5]

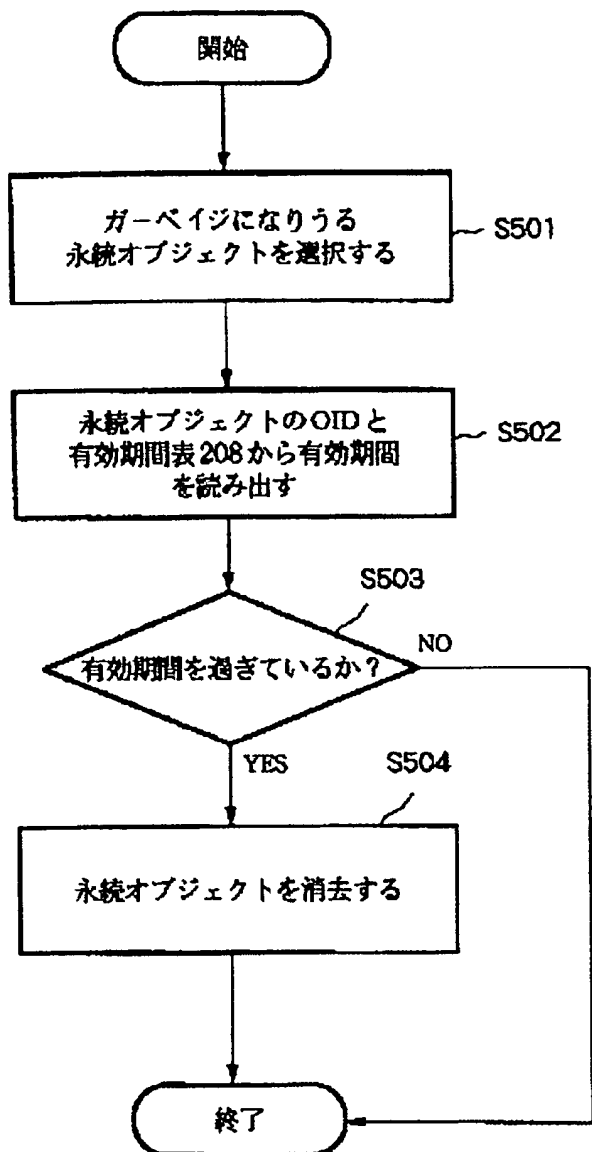
[Document to be Amended] DRAWINGS

[Item(s) to be Amended] drawing 5

[Method of Amendment] Modification

[Proposed Amendment]

[Drawing 5]



[Translation done.]